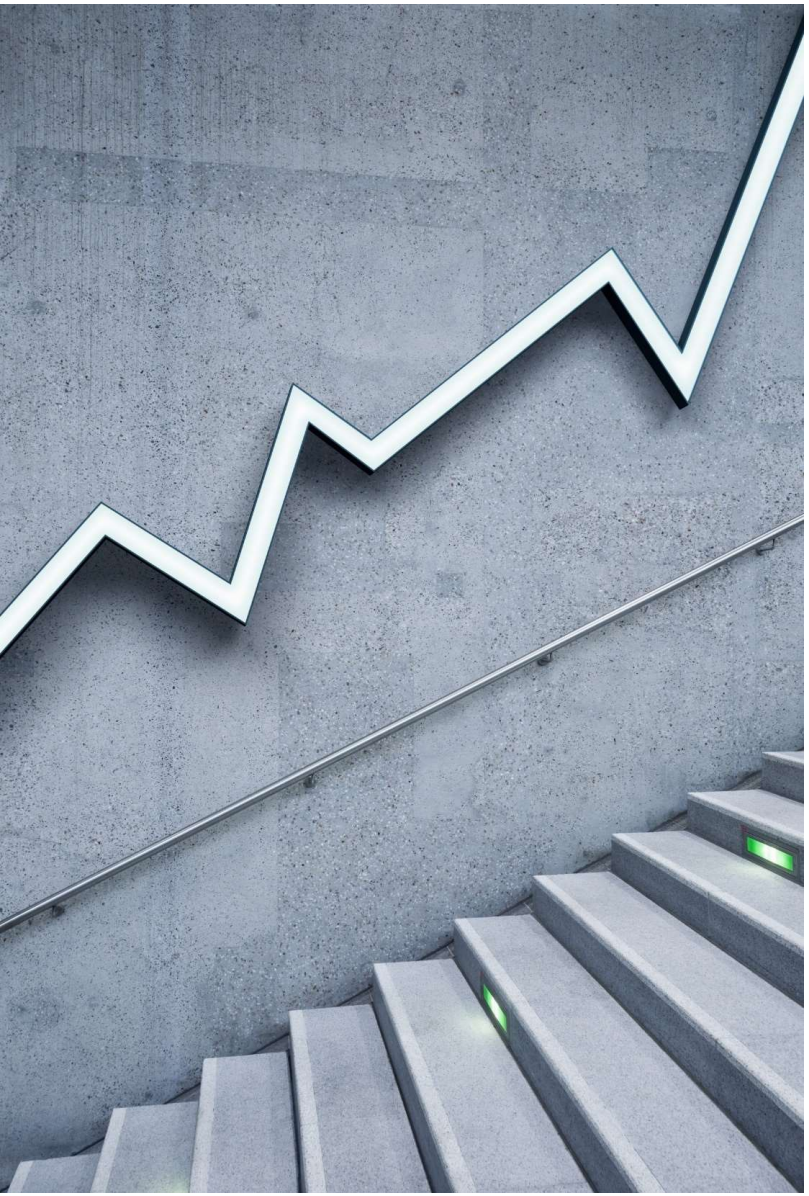

Learning Python

Praveen Nair

Introduction to Python

1. Invented in the Netherlands, early 90s by Guido van Rossum named after Monty Python Flying Circus series
2. Open sourced from the beginning
3. Scalable, object oriented and functional from the beginning
4. Increasingly popular
5. Strong Libraries

What is Python used for?

1. Data Analysis and Machine Learning
2. Web & Desktop Application Development
3. Automation or Scripting
4. Scientific and Numeric Applications
5. Beginner-friendly programming language
6. Game Development
7. Programming Internet of Things (IoT) devices

Installing Python...

1. Visit https://www.python.org/downloads/
2. Click on Download Python button
3. Run the installer (make sure to tick add path checkbox)

Installing Visual Studio Code...

1. Visit https://code.visualstudio.com/download
2. Click on Download button
3. Run the installer

Variables

We use variables to temporarily store data in computer's memory.

price = 10 (Integer) #type(price)

rating = 4.9 (Python represent float values as 64-bit "double-precision" values so no separate type as double)

course_name = 'Python for Beginners' (String)

is_published = True (Boolean)

Data Structures – List, Tuple, Dictionary, Set, Array (A data structure is a specialized format for organizing, processing, retrieving and storing data.)

……

Type function, multiple assignment, variable naming rule

Printing Variables

- `name="John"`
- `print(name)`

Printing with end parameter

- By default, the print function ends with a newline.
- Passing the whitespace to the end parameter (end=‘ ‘) indicates that the end character has to be identified by whitespace and not a newline.
- ----------
- print("Hello", end=' ')
- print("World")

Receiving Input

- `name=input("Enter your name")`
- `print(name)`

Type Conversion

- `birth_year = input("Enter your birth year")`
- `age = 2022 - int(birth_year)`
- `print(age)`
- `---------------`
- `float()`
- `Eval()`

Comments

- `birth_year = input("Enter your birth year")`
- `age = 2022 - int(birth_year)`
- `# print(age)`
- `Print(birth_year)`
- `---------`
- How to comment multiple lines?

Arithmetic Operations

- `+`
- `-`
- `*`
- `/  returns a float`
- `//  returns an int`
- `%  returns the remainder of division`
- `** exponentiation`

Comparison Operators in Python

- `Equals: a == b`
- `Not Equals: a != b`
- `Less than: a < b`
- `Less than or equal to: a <= b`
- `Greater than: a > b`
- `Greater than or equal to: a >= b`

Logical Operators in Python

- `and: (a < b) and (a < c)`
- `or: (a < b) or (a < c)`
- `Not: not(a > c)`

Strings

- We can define strings using single or double quotes.
- msg = "This is a dummy message"
- print(msg)
- ---------------
- msg = 'Hello "John"'
- print(msg)
- ---------------
- How to add apostrophe in a string?

Multi-line String

- To define a multi-line string, we surround our string with tripe quotes.
- msg = '''
- Dear Friend,
- How are you? Hope all is well with you.
- Thanks,
- MyName
- '''
- print(msg)

Concatenate Strings

- `first_name = "John "`
- `last_name = "Smith "`
- `print(first_name + last_name)`

Formatted String Literals (f-string)

- `name=input("Enter Name ")`
- `favorite_color=input("Enter Favorite Colour ")`
- `msg = f'{name} likes {favorite_color}’`
- `print(msg)`

Python Escape Characters

- `txt="This is John\'s book." # \' means '`
- `print(txt)`
- `txt="This is a \n dummy text." # new line`
- `print(txt)`
- `txt="Name \t City \t Age" # tabspace`
- `print(txt)`

if…statement

```
a=10
b=15
if(a>b):
    print("a is big")
else:
    print("b is big")
```

if…elif…statement

```
a=10
b=10
if(a>b):
    print("a is greater")
elif(b>a):
    print("b is greater")
else:
    print("Both a and b are equal")
```